**Part A of the Project**

--For avg acceptane rate of super\_host vs other\_host For City Athens

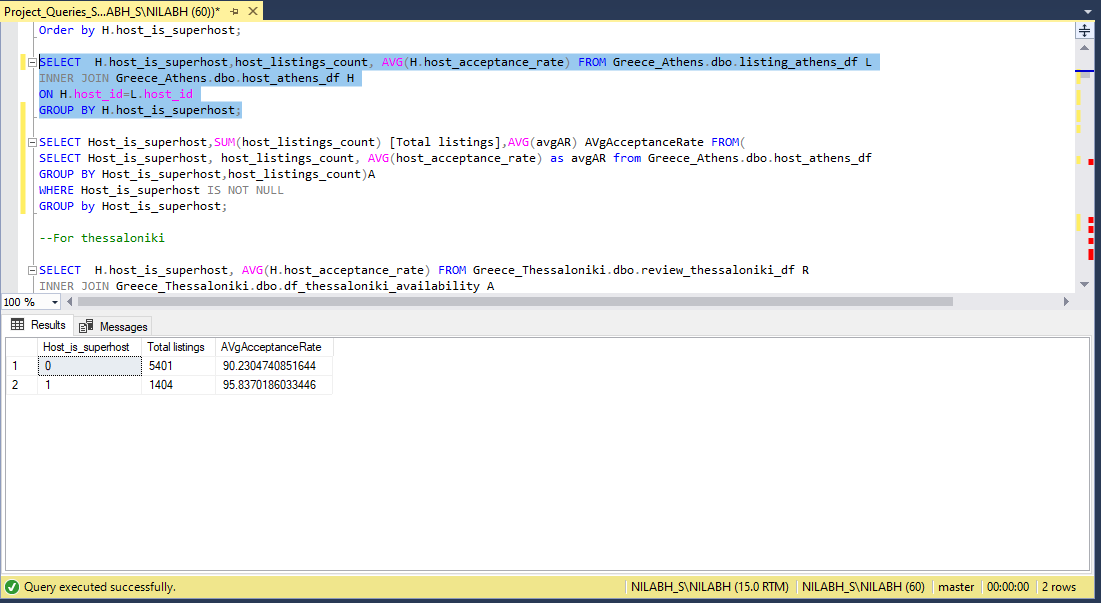
SELECT Host\_is\_superhost,SUM(host\_listings\_count) [Total listings],AVG(avgAR) AVgAcceptanceRate FROM(

SELECT Host\_is\_superhost, host\_listings\_count, AVG(host\_acceptance\_rate) as avgAR from Greece\_Athens.dbo.host\_athens\_df

GROUP BY Host\_is\_superhost,host\_listings\_count)A

WHERE Host\_is\_superhost IS NOT NULL

GROUP by Host\_is\_superhost;



--For avg acceptane rate of super\_host vs other\_host For City Thessaloniki

SELECT Host\_is\_superhost,SUM(host\_listings\_count) [Total listings],

AVG(avgAR) AVgAcceptanceRate FROM(

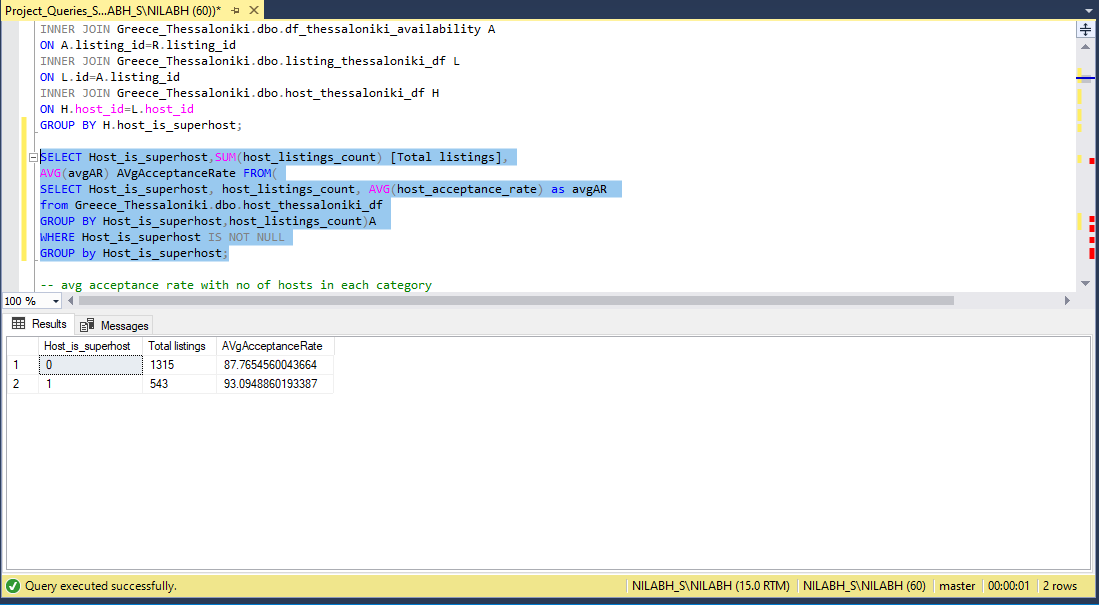
SELECT Host\_is\_superhost, host\_listings\_count, AVG(host\_acceptance\_rate) as avgAR

from Greece\_Thessaloniki.dbo.host\_thessaloniki\_df

GROUP BY Host\_is\_superhost,host\_listings\_count)A

WHERE Host\_is\_superhost IS NOT NULL

GROUP by Host\_is\_superhost;



--No of hosts that Provide instant bookable in each category

--For Athens

SELECT H.host\_is\_superhost, H.host\_response\_time,AVG(H.host\_acceptance\_rate) AVG\_Acceptance\_Rate, COUNT(H.host\_is\_superhost) Host\_count,

SUM(CAST(L.instant\_bookable AS INT)) Host\_Provide\_instant\_booking

FROM Greece\_Athens.dbo.host\_athens\_df H

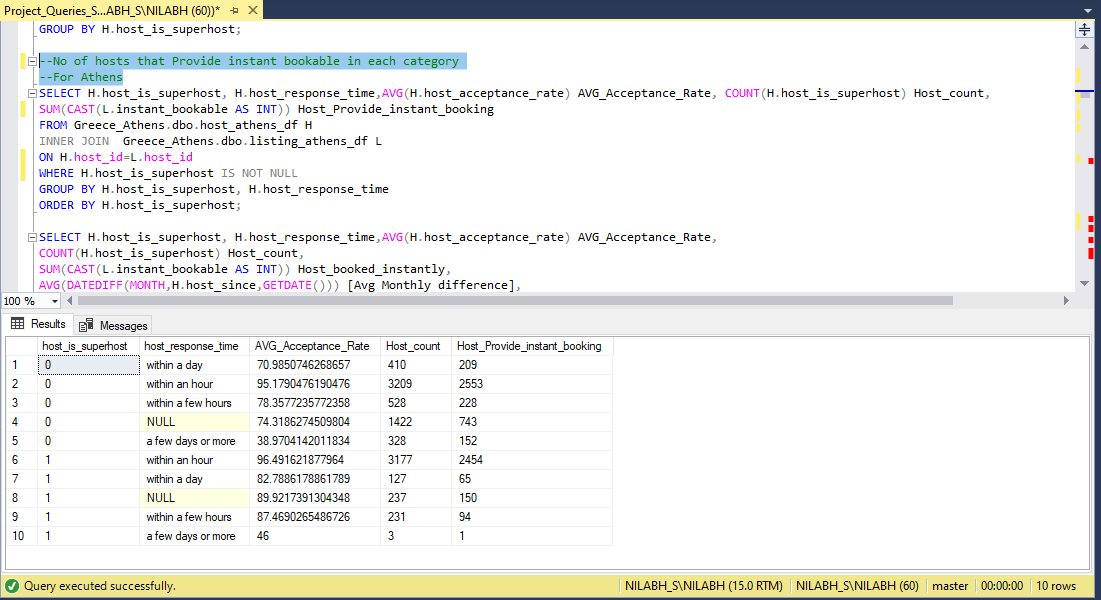
INNER JOIN Greece\_Athens.dbo.listing\_athens\_df L

ON H.host\_id=L.host\_id

WHERE H.host\_is\_superhost IS NOT NULL

GROUP BY H.host\_is\_superhost, H.host\_response\_time

ORDER BY H.host\_is\_superhost;



--No of hosts that Provide instant bookable in each category with age of host in months since joined

SELECT H.host\_is\_superhost, H.host\_response\_time,AVG(H.host\_acceptance\_rate) AVG\_Acceptance\_Rate,

COUNT(H.host\_is\_superhost) Host\_count,

SUM(CAST(L.instant\_bookable AS INT)) Host\_booked\_instantly,

AVG(DATEDIFF(MONTH,H.host\_since,GETDATE())) [Avg Monthly difference],

AVG(DATEDIFF(YEAR,H.host\_since,GETDATE())) [Avg Yearly difference]

FROM Greece\_Athens.dbo.host\_athens\_df H

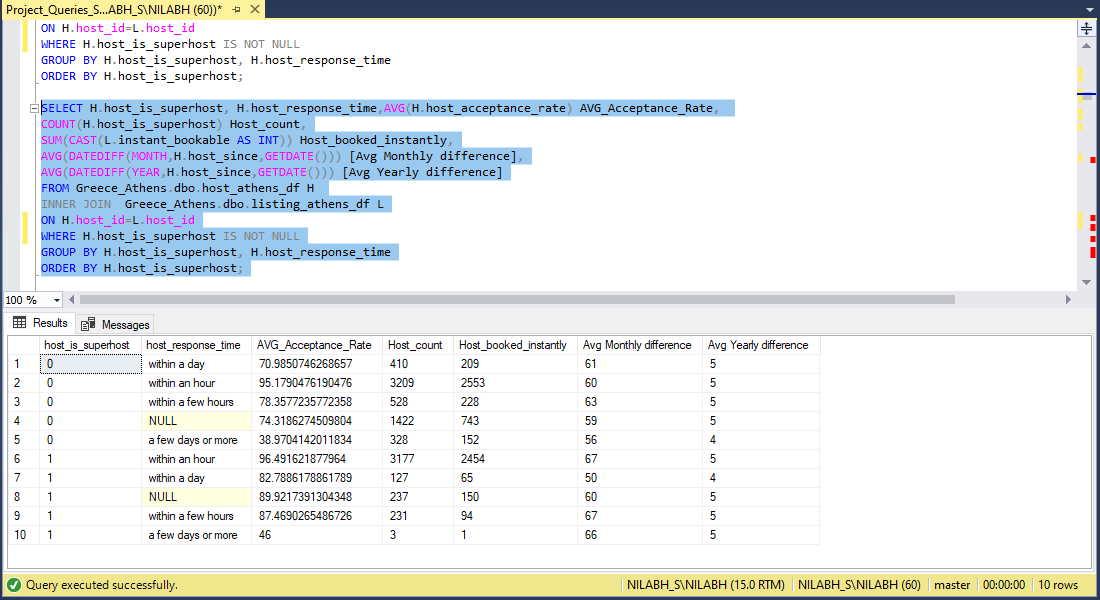
INNER JOIN Greece\_Athens.dbo.listing\_athens\_df L

ON H.host\_id=L.host\_id

WHERE H.host\_is\_superhost IS NOT NULL

GROUP BY H.host\_is\_superhost, H.host\_response\_time

ORDER BY H.host\_is\_superhost;



--No of hosts that Provide instant bookable in each category

--For Thessaloniki

SELECT H.host\_is\_superhost, H.host\_response\_time,AVG(H.host\_acceptance\_rate) AVG\_Acceptance\_Rate,

COUNT(H.host\_is\_superhost) Host\_count,

SUM(CAST(L.instant\_bookable AS INT)) Host\_booked\_instantly

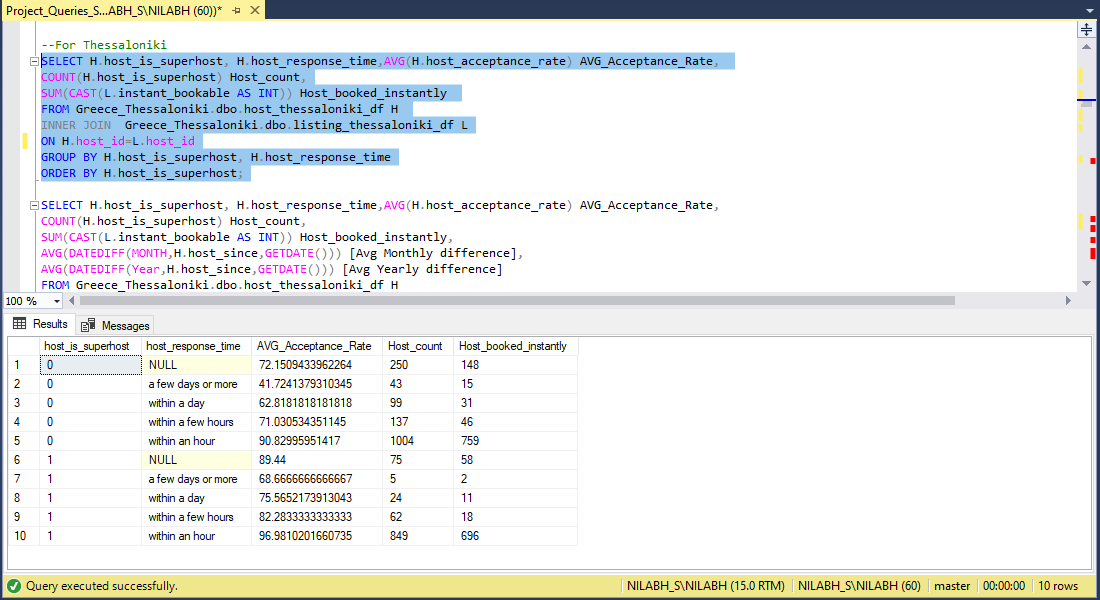
FROM Greece\_Thessaloniki.dbo.host\_thessaloniki\_df H

INNER JOIN Greece\_Thessaloniki.dbo.listing\_thessaloniki\_df L

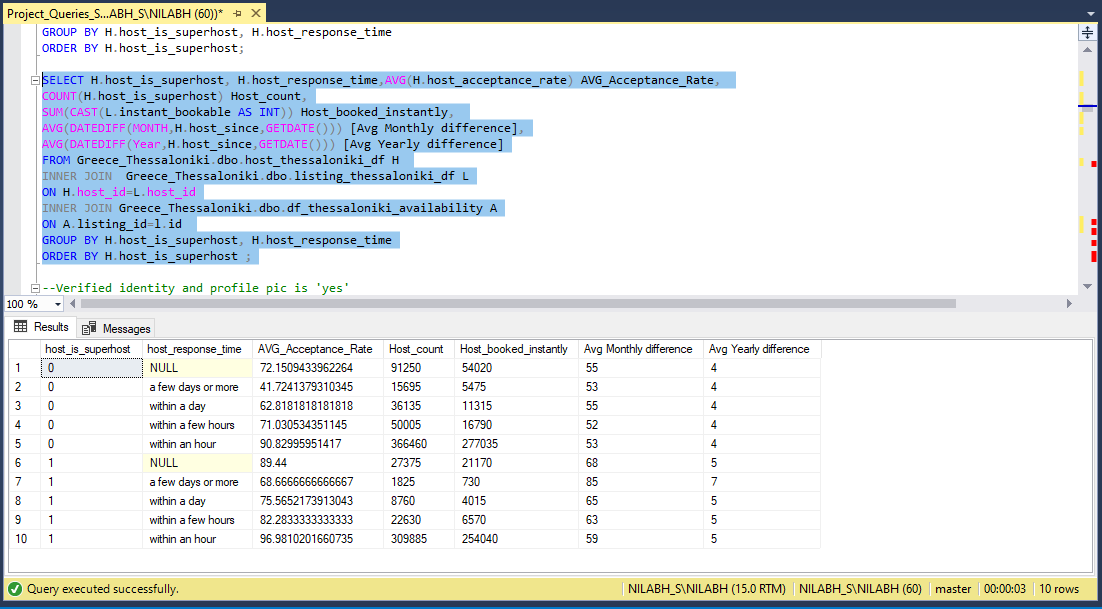
ON H.host\_id=L.host\_id

GROUP BY H.host\_is\_superhost, H.host\_response\_time

ORDER BY H.host\_is\_superhost;



--No of hosts that Provide instant bookable in each category with age of host in months since joined



--Verified identity and profile pic is 'yes'

--For Athens

SELECT H.host\_is\_superhost, COUNT(host\_is\_superhost) #OfHosts,

SUM(host\_listings\_count) Host\_holds\_listing,

sum(CAST(H.host\_identity\_verified AS INT)) [Verified Identity],

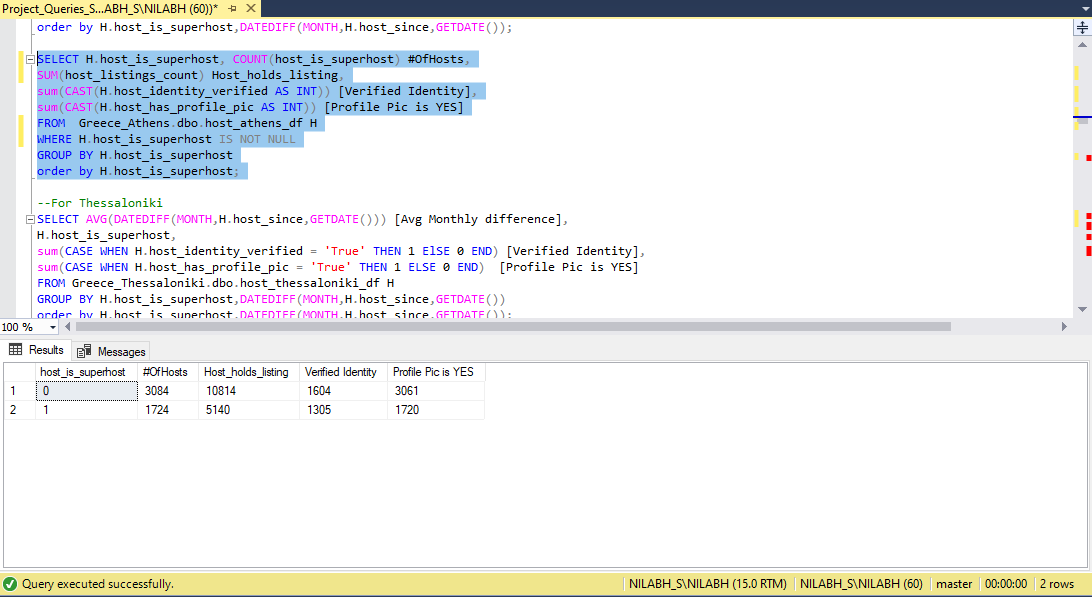
sum(CAST(H.host\_has\_profile\_pic AS INT)) [Profile Pic is YES]

FROM Greece\_Athens.dbo.host\_athens\_df H

WHERE H.host\_is\_superhost IS NOT NULL

GROUP BY H.host\_is\_superhost

order by H.host\_is\_superhost;



--Verified identity and profile pic is 'yes'

--For Thessaloniki

SELECT H.host\_is\_superhost, COUNT(host\_is\_superhost) #OfHosts,

SUM(host\_listings\_count) Host\_holds\_listing,

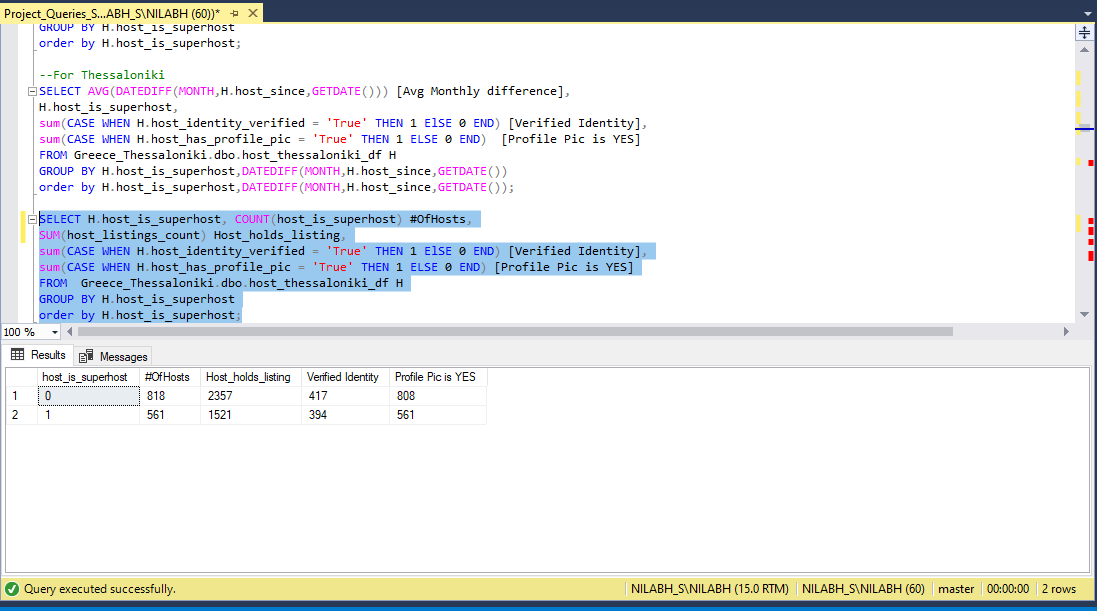
sum(CASE WHEN H.host\_identity\_verified = 'True' THEN 1 ElSE 0 END) [Verified Identity],

sum(CASE WHEN H.host\_has\_profile\_pic = 'True' THEN 1 ELSE 0 END) [Profile Pic is YES]

FROM Greece\_Thessaloniki.dbo.host\_thessaloniki\_df H

GROUP BY H.host\_is\_superhost

order by H.host\_is\_superhost;



--AVG ratings for hosts according to year of joining the company

--For Athens

SELECT H.host\_is\_superhost,COUNT(H.host\_is\_superhost) #ofHosts,

DATEPART(YEAR,H.host\_since) [Host joining year],COUNT(L.host\_id) [listing belongs to host] ,

ROUND(AVG(L.review\_scores\_accuracy),2) R\_Accuracy,ROUND(AVG(L.review\_scores\_checkin),2) R\_Checkin,

ROUND(AVG(L.review\_scores\_cleanliness),2) R\_Cleanliness,ROUND(AVG(L.review\_scores\_communication),2) R\_Communication,

ROUND(AVG(L.review\_scores\_location),2) R\_location,

ROUND(AVG(L.review\_scores\_rating),2) R\_Rating,

ROUND(AVG(L.review\_scores\_value),2) R\_Value,

ROUND(AVG(H.host\_response\_rate),2) Host\_Response\_Rate

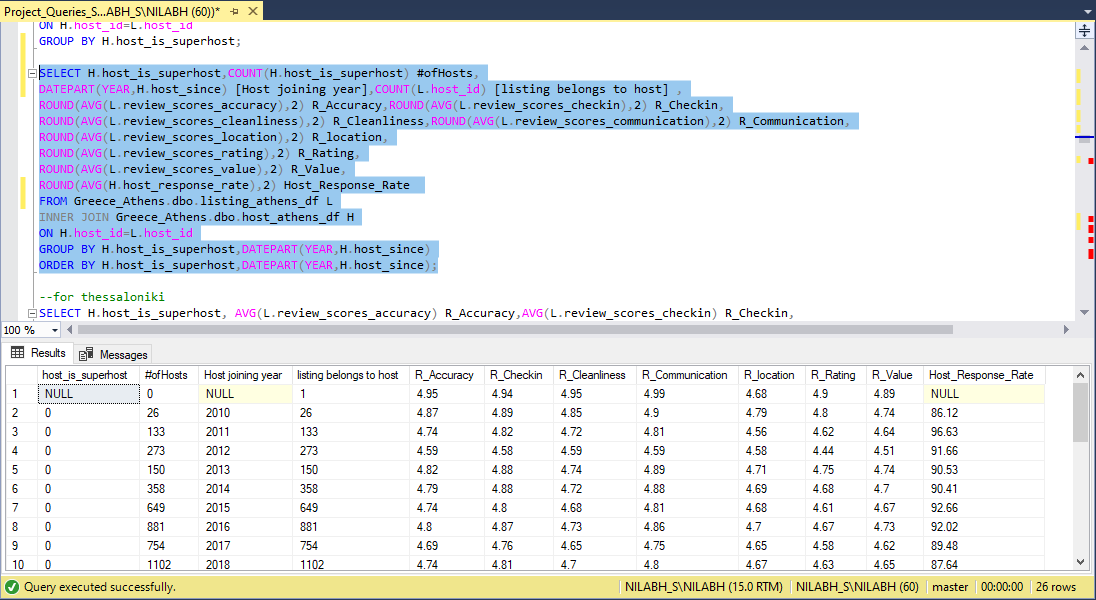
FROM Greece\_Athens.dbo.listing\_athens\_df L

INNER JOIN Greece\_Athens.dbo.host\_athens\_df H

ON H.host\_id=L.host\_id

GROUP BY H.host\_is\_superhost,DATEPART(YEAR,H.host\_since)

ORDER BY H.host\_is\_superhost,DATEPART(YEAR,H.host\_since);



--AVG ratings for hosts according to year of joining the company

--For Thessaloniki

SELECT H.host\_is\_superhost,DATEPART(YEAR,H.host\_since) [Host joining year],

COUNT(L.host\_id) [listing belongs to host] ,

ROUND(AVG(L.review\_scores\_accuracy),2) R\_Accuracy,

ROUND(AVG(L.review\_scores\_checkin),2) R\_Checkin,

ROUND(AVG(L.review\_scores\_cleanliness),2) R\_Cleanliness,

ROUND(AVG(L.review\_scores\_communication),2) R\_Communication,

ROUND(AVG(L.review\_scores\_location),2) R\_location,

ROUND(AVG(L.review\_scores\_rating),2) R\_Rating,

ROUND(AVG(L.review\_scores\_value),2) R\_Value,

ROUND(AVG(H.host\_response\_rate),2) Host\_Response\_Rate

FROM Greece\_Thessaloniki.dbo.review\_thessaloniki\_df R

INNER JOIN Greece\_Thessaloniki.dbo.listing\_thessaloniki\_df L

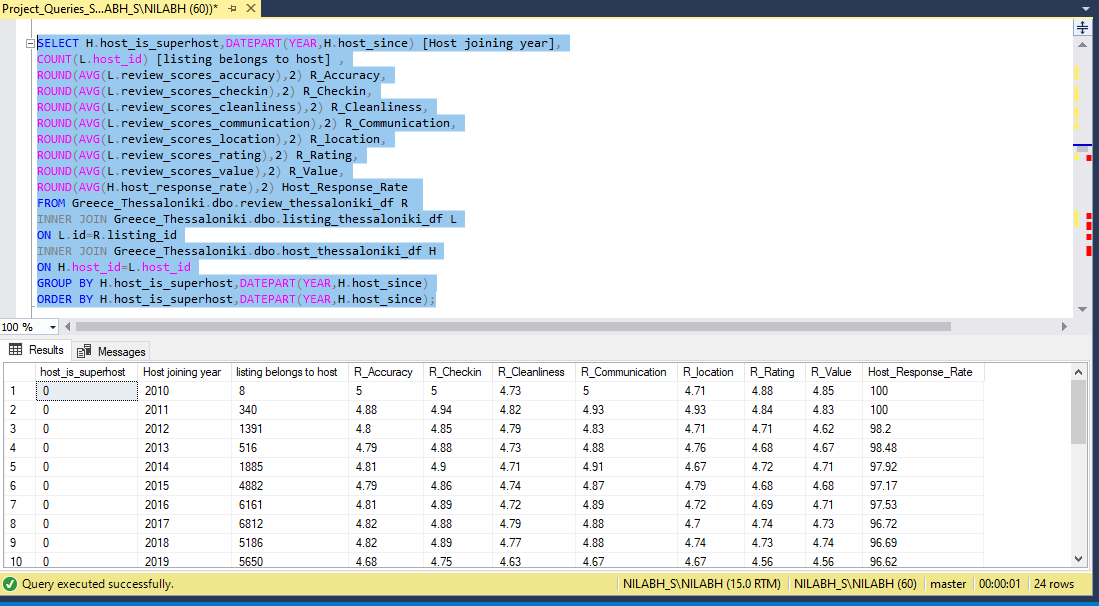
ON L.id=R.listing\_id

INNER JOIN Greece\_Thessaloniki.dbo.host\_thessaloniki\_df H

ON H.host\_id=L.host\_id

GROUP BY H.host\_is\_superhost,DATEPART(YEAR,H.host\_since)

ORDER BY H.host\_is\_superhost,DATEPART(YEAR,H.host\_since);



--AVG ratings for hosts with host since joined

--For athens

SELECT DATEDIFF(YEAR,H.host\_since,GETDATE()) [Age of Host WRT Company],H.host\_is\_superhost,

ROUND(AVG(L.review\_scores\_accuracy),2) R\_Accuracy,

ROUND(AVG(L.review\_scores\_checkin),2) R\_Checkin,

ROUND(AVG(L.review\_scores\_cleanliness),2) R\_Cleanliness,

ROUND(AVG(L.review\_scores\_communication),2) R\_Communication,

ROUND(AVG(L.review\_scores\_location),2) R\_location,

ROUND(AVG(L.review\_scores\_rating),2) R\_Rating,

ROUND(AVG(L.review\_scores\_value),2) R\_Value,

ROUND(AVG(H.host\_response\_rate),2) Avg\_response\_rate

FROM Greece\_Athens.dbo.df\_athens\_availability A

INNER JOIN Greece\_Athens.dbo.listing\_athens\_df L

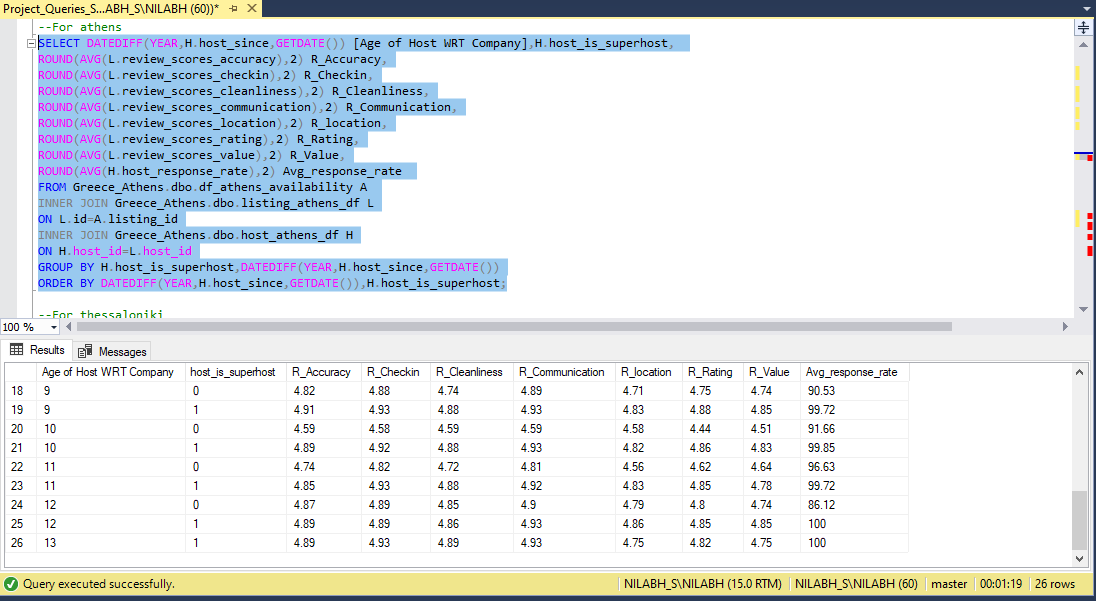
ON L.id=A.listing\_id

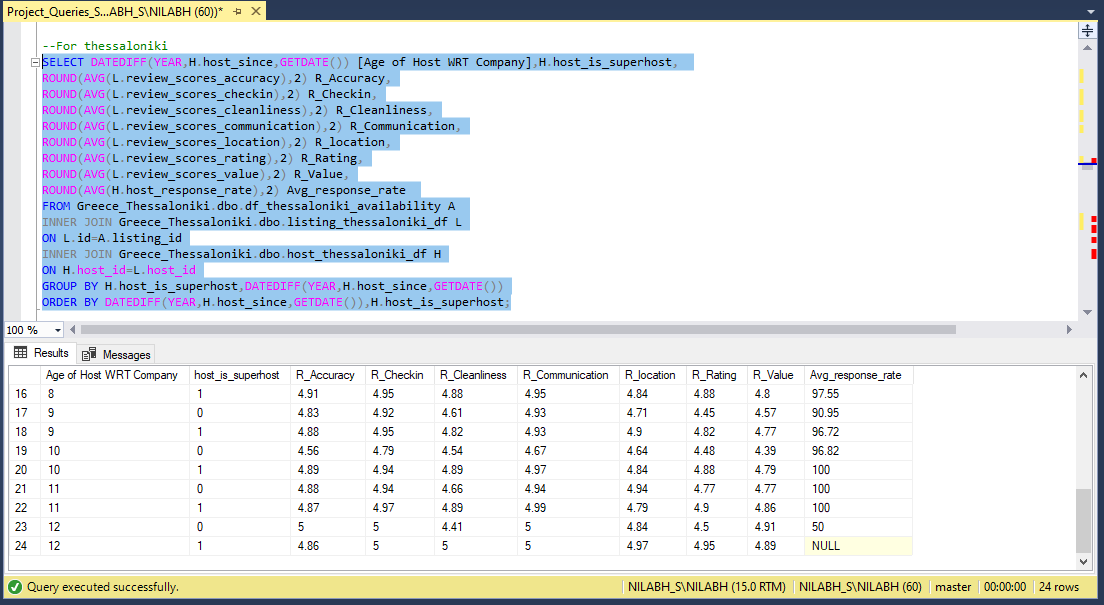
INNER JOIN Greece\_Athens.dbo.host\_athens\_df H

ON H.host\_id=L.host\_id

GROUP BY H.host\_is\_superhost,DATEDIFF(YEAR,H.host\_since,GETDATE())

ORDER BY DATEDIFF(YEAR,H.host\_since,GETDATE()),H.host\_is\_superhost;





--AVG no of booking per month for all cases  
--For Athens

SELECT [year],host\_is\_superhost,SUM([NoofBokking]) NoOfBooking FROM(

SELECT DATENAME(YEAR,H.host\_since) [year],H.host\_is\_superhost,

COUNT(R.reviewer\_id) [NoofBokking]

FROM Greece\_Athens.dbo.review\_athens\_df R

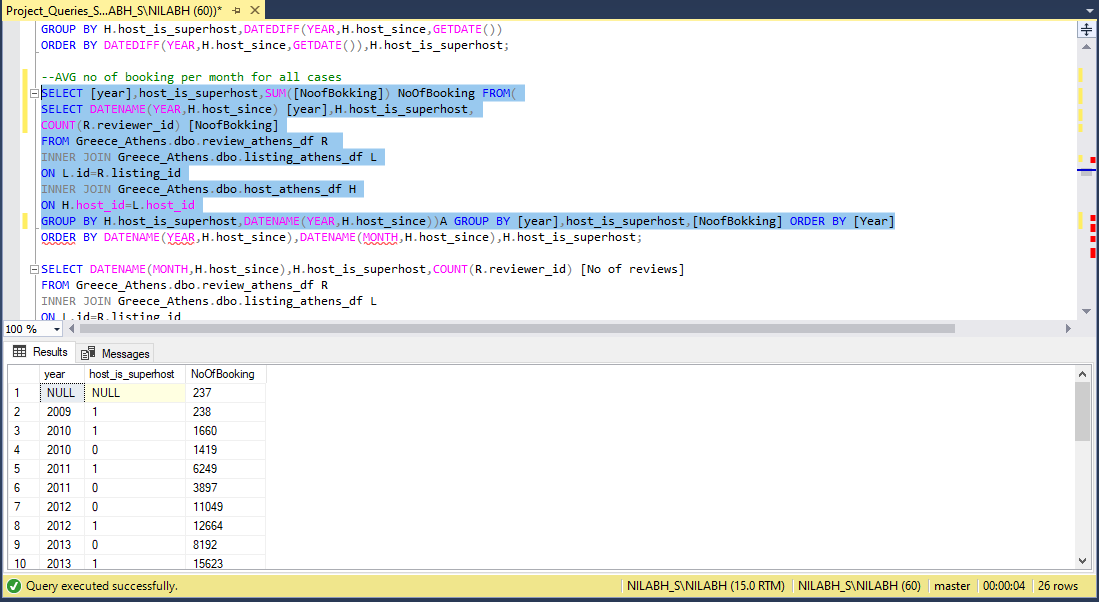
INNER JOIN Greece\_Athens.dbo.listing\_athens\_df L

ON L.id=R.listing\_id

INNER JOIN Greece\_Athens.dbo.host\_athens\_df H

ON H.host\_id=L.host\_id

GROUP BY H.host\_is\_superhost,DATENAME(YEAR,H.host\_since))A GROUP BY [year],host\_is\_superhost,[NoofBokking] ORDER BY [Year]



--For Thessaloniki (Month wise and year)

SELECT DATENAME(MONTH,H.host\_since),DATENAME(YEAR,H.host\_since),

H.host\_is\_superhost,

COUNT(R.reviewer\_id) [No of reviews]

FROM Greece\_Thessaloniki.dbo.review\_thessaloniki\_df R

INNER JOIN Greece\_Thessaloniki.dbo.listing\_thessaloniki\_df L

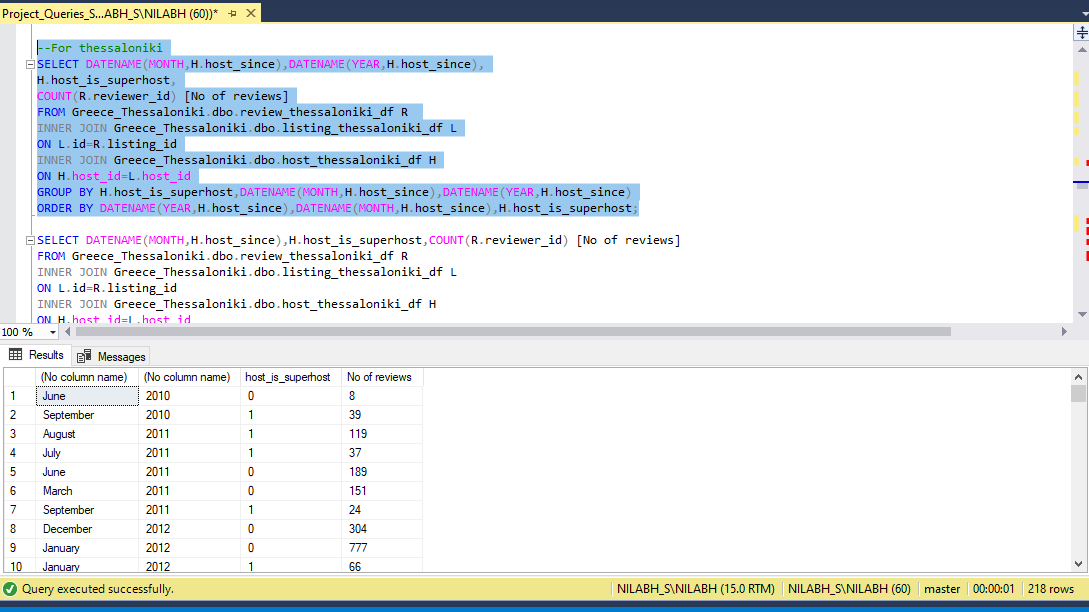
ON L.id=R.listing\_id

INNER JOIN Greece\_Thessaloniki.dbo.host\_thessaloniki\_df H

ON H.host\_id=L.host\_id

GROUP BY H.host\_is\_superhost,DATENAME(MONTH,H.host\_since),DATENAME(YEAR,H.host\_since)

ORDER BY DATENAME(YEAR,H.host\_since),DATENAME(MONTH,H.host\_since),H.host\_is\_superhost;



--c. Analyze how does the comments of reviewers vary for listings of Super Hosts vs Other Hosts

--(Extract words from the comments provided by the reviewers)  
--For Athens

SELECT H.host\_is\_superhost,COUNT(CASE WHEN CHARINDEX('great', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_great,

COUNT(CASE WHEN CHARINDEX('Amazing', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_Amazing,

COUNT(CASE WHEN CHARINDEX('Friendly', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_Friendly,

COUNT(CASE WHEN CHARINDEX('recommend', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_recommend,

COUNT(CASE WHEN CHARINDEX(H.host\_neighbourhood, R.comments,1)>0 THEN 1 ELSE NULL END) Count\_host\_neighbourhood,

COUNT(CASE WHEN CHARINDEX('helpful', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_helpful,

COUNT(CASE WHEN CHARINDEX(H.host\_name, R.comments,1)>0 THEN 1 ELSE NULL END) Count\_host\_name,

COUNT(CASE WHEN CHARINDEX('very nice', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_very\_nice,

COUNT(CASE WHEN CHARINDEX('home', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_home,

COUNT(CASE WHEN CHARINDEX('decorated', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_decorated,

COUNT(CASE WHEN CHARINDEX('refurbished', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_refurbished,

COUNT(CASE WHEN CHARINDEX('Wonderful', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_Wonderful,

COUNT(CASE WHEN CHARINDEX('comfortable', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_comfortable

FROM Greece\_Athens.dbo.review\_athens\_df R

INNER JOIN Greece\_Athens.dbo.listing\_athens\_df L

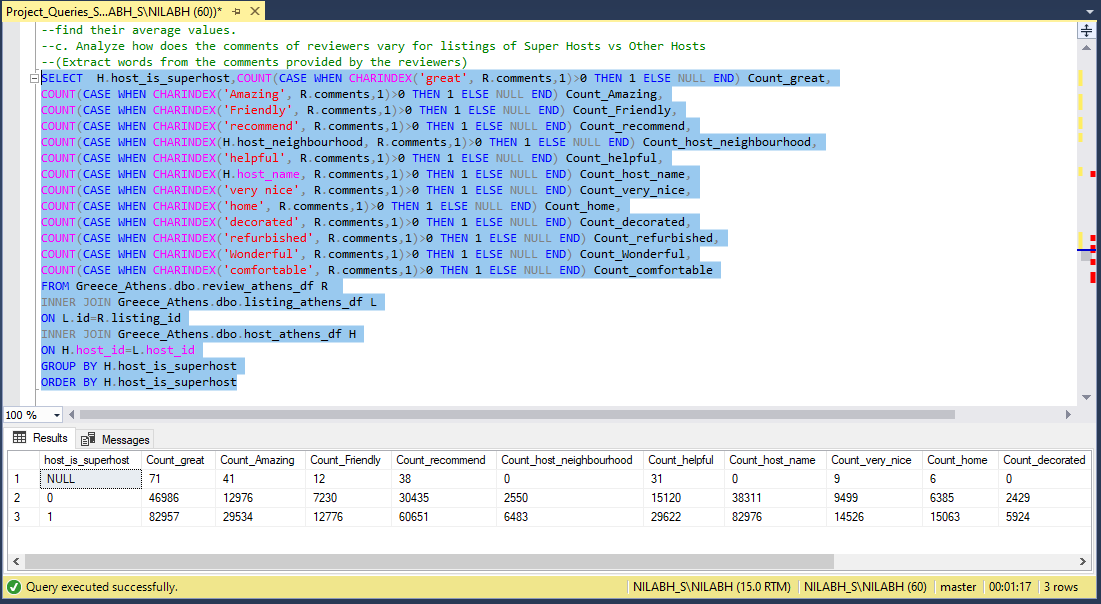
ON L.id=R.listing\_id

INNER JOIN Greece\_Athens.dbo.host\_athens\_df H

ON H.host\_id=L.host\_id

GROUP BY H.host\_is\_superhost

ORDER BY H.host\_is\_superhost



-- For Thessaloniki

SELECT H.host\_is\_superhost,

COUNT(CASE WHEN CHARINDEX('great', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_great,

COUNT(CASE WHEN CHARINDEX('Amazing', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_Amazing,

COUNT(CASE WHEN CHARINDEX('Friendly', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_Friendly,

COUNT(CASE WHEN CHARINDEX('recommend', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_recommend,

COUNT(CASE WHEN CHARINDEX(H.host\_neighbourhood, R.comments,1)>0 THEN 1 ELSE NULL END) Count\_host\_neighbourhood,

COUNT(CASE WHEN CHARINDEX('helpful', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_helpful,

COUNT(CASE WHEN CHARINDEX(H.host\_name, R.comments,1)>0 THEN 1 ELSE NULL END) Count\_host\_name,

COUNT(CASE WHEN CHARINDEX('very nice', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_very\_nice,

COUNT(CASE WHEN CHARINDEX('home', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_home,

COUNT(CASE WHEN CHARINDEX('decorated', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_decorated,

COUNT(CASE WHEN CHARINDEX('refurbished', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_refurbished,

COUNT(CASE WHEN CHARINDEX('Wonderful', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_Wonderful,

COUNT(CASE WHEN CHARINDEX('comfortable', R.comments,1)>0 THEN 1 ELSE NULL END) Count\_comfortable

FROM Greece\_Thessaloniki.dbo.review\_thessaloniki\_df R

INNER JOIN Greece\_Thessaloniki.dbo.listing\_thessaloniki\_df L

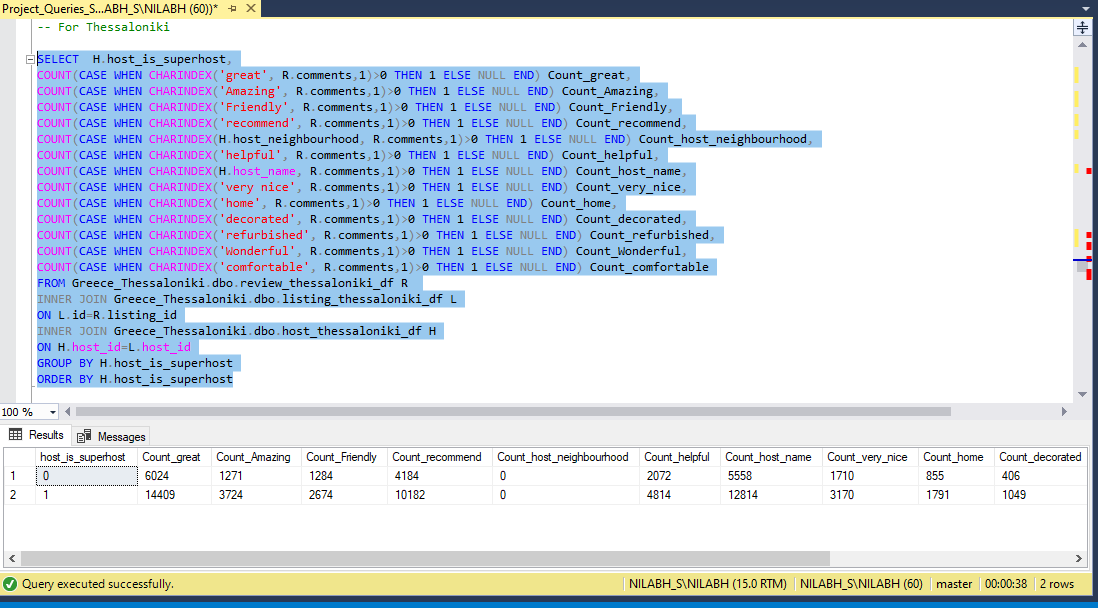
ON L.id=R.listing\_id

INNER JOIN Greece\_Thessaloniki.dbo.host\_thessaloniki\_df H

ON H.host\_id=L.host\_id

GROUP BY H.host\_is\_superhost

ORDER BY H.host\_is\_superhost



--e. Analyze the average price and availability of the listings for

--the upcoming year between Super Hosts and Other Hosts

--FOR Athens

SELECT [YEAR],id,available,AVG(AvgAdjustPrice) OVER (Partition by id,available Order by avgAdjustprice) FROM (

SELECT H.host\_is\_superhost,A.available,L.id,

DATEPART(YEAR,CAST(A.[date] AS date)) [Year], AVG(L.price) AvgAdjustPrice

from Greece\_Athens.dbo.df\_athens\_availability A

INNER JOIN Greece\_Athens.dbo.listing\_athens\_df L

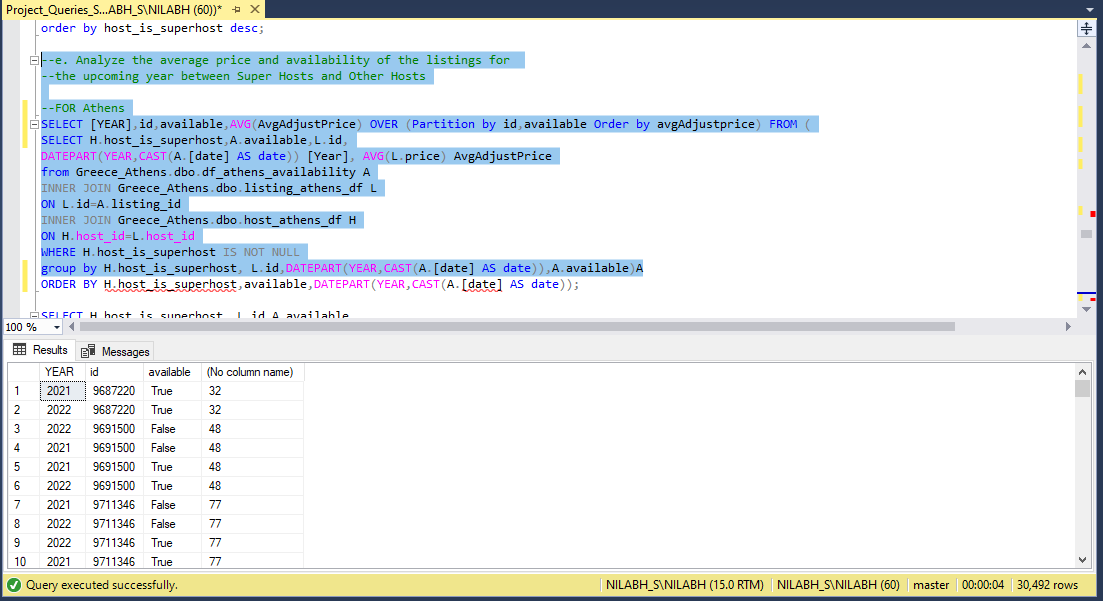
ON L.id=A.listing\_id

INNER JOIN Greece\_Athens.dbo.host\_athens\_df H

ON H.host\_id=L.host\_id

WHERE H.host\_is\_superhost IS NOT NULL

group by H.host\_is\_superhost, L.id,DATEPART(YEAR,CAST(A.[date] AS date)),A.available)A



SELECT H.host\_is\_superhost,A.available,

DATEPART(YEAR,CAST(A.[date] AS date)) [Year], AVG(L.price) AvgAdjustPrice

from Greece\_Athens.dbo.df\_athens\_availability A

INNER JOIN Greece\_Athens.dbo.listing\_athens\_df L

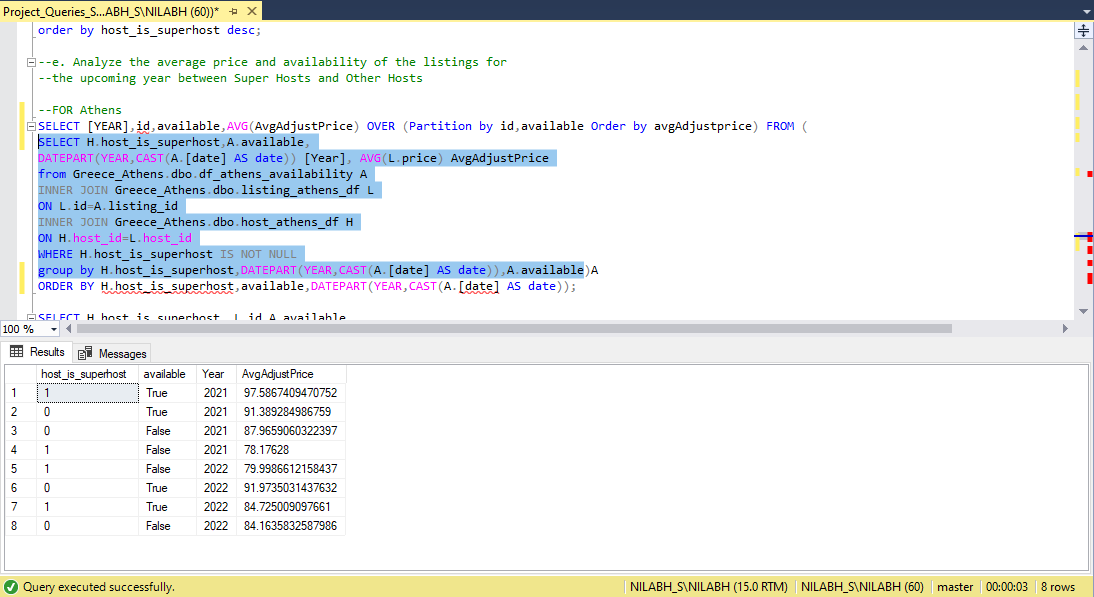
ON L.id=A.listing\_id

INNER JOIN Greece\_Athens.dbo.host\_athens\_df H

ON H.host\_id=L.host\_id

WHERE H.host\_is\_superhost IS NOT NULL

group by H.host\_is\_superhost,DATEPART(YEAR,CAST(A.[date] AS date)),A.available



--For Thessaloniki

SELECT host\_is\_superhost,[YEAR],id,available,AVG(AvgAdjustPrice) OVER (Partition by id,available Order by avgAdjustprice) FROM (

SELECT H.host\_is\_superhost, L.id,A.available,

DATEPART(YEAR,CAST(A.[date] AS date)) [Year], AVG(L.price) AvgAdjustPrice

from Greece\_Thessaloniki.dbo.df\_thessaloniki\_availability A

INNER JOIN Greece\_Thessaloniki.dbo.listing\_thessaloniki\_df L

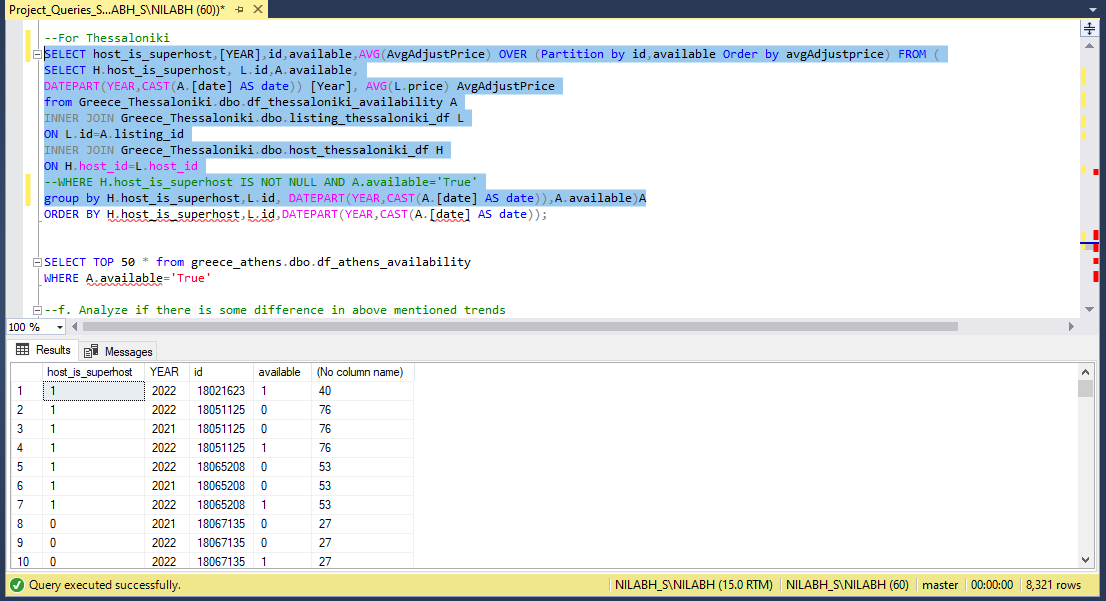
ON L.id=A.listing\_id

INNER JOIN Greece\_Thessaloniki.dbo.host\_thessaloniki\_df H

ON H.host\_id=L.host\_id

--WHERE H.host\_is\_superhost IS NOT NULL AND A.available='True'

group by H.host\_is\_superhost,L.id, DATEPART(YEAR,CAST(A.[date] AS date)),A.available)A



SELECT H.host\_is\_superhost, A.available,

DATEPART(YEAR,CAST(A.[date] AS date)) [Year], AVG(L.price) AvgAdjustPrice

from Greece\_Thessaloniki.dbo.df\_thessaloniki\_availability A

INNER JOIN Greece\_Thessaloniki.dbo.listing\_thessaloniki\_df L

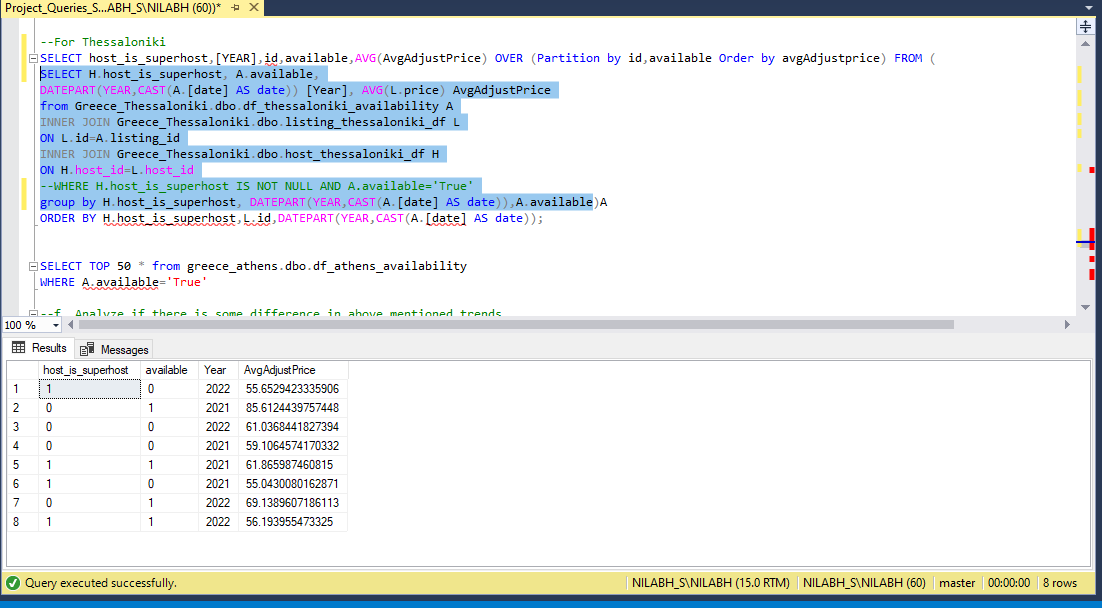
ON L.id=A.listing\_id

INNER JOIN Greece\_Thessaloniki.dbo.host\_thessaloniki\_df H

ON H.host\_id=L.host\_id

--WHERE H.host\_is\_superhost IS NOT NULL AND A.available='True'

group by H.host\_is\_superhost, DATEPART(YEAR,CAST(A.[date] AS date)),A.available



--f. Analyze if there is some difference in above mentioned trends

--between Local Hosts or Hosts residing in other locations

SELECT top 500 \*,

SUM(CASE WHEN (neighbourhood\_cleansed= 'ΑΓΙΟΣ ΝΙΚΟΛΑΟΣ' AND host\_neighbourhood = 'Agios Nikolaos') THEN 1 ELSE 0 END),

CASE WHEN (neighbourhood\_cleansed= 'ΑΚΑΔΗΜΙΑ ΠΛΑΤΩΝΟΣ' AND host\_neighbourhood = 'Akadimia Platonos') THEN 1 END,

CASE WHEN (neighbourhood\_cleansed= 'ΑΜΠΕΛΟΚΗΠΟΙ' AND host\_neighbourhood = 'Ampelokipoi') THEN 1 END,

CASE WHEN (neighbourhood\_cleansed= 'ΑΜΠΕΛΟΚΗΠΟΙ' AND host\_neighbourhood = 'Ambelokipi') THEN 1 END,

CASE WHEN (neighbourhood\_cleansed= 'ΙΛΙΣΙΑ' AND host\_neighbourhood = 'Ilisia') THEN 1 END,

CASE WHEN (neighbourhood\_cleansed= 'ΚΟΛΩΝΑΚΙ' AND host\_neighbourhood = 'Kolonaki') THEN 1 END,

CASE WHEN (neighbourhood\_cleansed= 'ΝΕΟΣ ΚΟΣΜΟΣ' AND host\_neighbourhood = 'Neos Kosmos') THEN 1 END,

CASE WHEN (neighbourhood\_cleansed= 'ΠΕΔΙΟ ΑΡΕΩΣ' AND host\_neighbourhood = 'Pedion Areos') THEN 1 END,

CASE WHEN (neighbourhood\_cleansed= 'ΠΑΤΗΣΙΑ' AND host\_neighbourhood = 'Patisia') THEN 1 END,

CASE WHEN (neighbourhood\_cleansed= 'ΠΕΤΡΑΛΩΝΑ' AND host\_neighbourhood = 'Petralona') THEN 1 END,

CASE WHEN (neighbourhood\_cleansed= 'ΣΕΠΟΛΙΑ' AND host\_neighbourhood = 'Sepolia') THEN 1 END

FROM(

SELECT H.host\_is\_superhost, L.id,A.available,L.neighbourhood\_cleansed,H.host\_neighbourhood,

DATEPART(YEAR,CAST(A.[date] AS date)) [Year], AVG(L.price) AvgAdjustPrice

from Greece\_Athens.dbo.df\_athens\_availability A

INNER JOIN Greece\_Athens.dbo.listing\_athens\_df L

ON L.id=A.listing\_id

INNER JOIN Greece\_Athens.dbo.host\_athens\_df H

ON H.host\_id=L.host\_id

--WHERE (L.neighbourhood\_cleansed= 'ΑΓΙΟΣ ΝΙΚΟΛΑΟΣ' AND H.host\_neighbourhood = 'Agios Nikolaos')

group by H.host\_is\_superhost,L.id, DATEPART(YEAR,CAST(A.[date] AS date)),A.available,L.neighbourhood\_cleansed,H.host\_neighbourhood)A

GROUP BY host\_is\_superhost,id,available,neighbourhood\_cleansed,host\_neighbourhood,[Year],AvgAdjustPrice

